



UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE  
INSTITUTO METRÓPOLE DIGITAL  
PROGRAMA DE RESIDÊNCIA EM TECNOLOGIA DA INFORMAÇÃO

# Desenvolvimento de modelo de Machine Learning para priorização de petições iniciais em processos judiciais

João Paulo de Oliveira Câmara Fernandes

Natal-RN, Brasil  
2023

João Paulo de Oliveira Câmara Fernandes

## Desenvolvimento de modelo de Machine Learning para priorização de petições iniciais em processos judiciais

Trabalho de Conclusão de Curso  
apresentado ao Programa de Residência em  
Tecnologia da Informação do Instituto  
Metrópole Digital da Universidade Federal  
do Rio Grande do Norte como requisito  
parcial para a obtenção do título de  
Especialista em Tecnologia da Informação.  
Área de Concentração: Business  
Intelligence & Analytics.

Orientador: Daniel Sabino Amorim de Araujo  
Coorientador: David Montalvão Junior

Natal-RN, Brasil  
2023

Universidade Federal do Rio Grande do Norte - UFRN  
Sistema de Bibliotecas - SISBI  
Catalogação de Publicação na Fonte. UFRN - Biblioteca Central Zila Mamede

Fernandes, João Paulo de Oliveira Câmara.

Desenvolvimento de modelo de Machine Learning para  
priorização de petições iniciais em processos judiciais / João  
Paulo de Oliveira Câmara Fernandes. - 2023.

48 f.: il.

Monografia (Especialização) - Universidade Federal do Rio  
Grande do Norte, Instituto Metrópole Digital, Programa de  
Residência em Tecnologia da Informação, Natal, RN, 2023.

Orientador: Prof. Dr. Daniel Sabino Amorim de Araujo.

Coorientador: Prof. Dr. David Montalvão Junior.

1. Machine learning - Monografia. 2. Processamento de  
linguagem natural - Monografia. 3. Petição inicial - Monografia.  
I. Araujo, Daniel Sabino Amorim de. II. Montalvão Junior, David.  
III. Título.

RN/UF/BCZM

CDU 004.62

João Paulo de Oliveira Câmara Fernandes

## Desenvolvimento de modelo de Machine Learning para priorização de petições iniciais em processos judiciais

Trabalho de Conclusão de Curso  
apresentado ao Programa de Residência em  
Tecnologia da Informação Aplicada à Área  
Jurídica do Instituto Metrópole Digital da  
Universidade Federal do Rio Grande do  
Norte como requisito parcial para a  
obtenção do título de Especialista em  
Tecnologia da Informação. Área de  
Concentração: Business Intelligence &  
Analytics.

Trabalho aprovado. Natal-RN, Brasil, Trinta e um de março de dois mil e vinte e três:

---

Daniel Sabino Amorim de Araujo  
Orientador

---

David Montalvão Junior  
Coorientador

---

Eduardo Henrique da Silva Aranha  
Examinador

Natal-RN, Brasil  
2023

*Dedicatória*

*Dedico este trabalho à minha família e à minha namorada, Dayana Guimarães, que foram muito importantes nessa minha trajetória.*

# Agradecimentos

Com imensa gratidão, dedico este trabalho às pessoas mais importantes da minha vida. Ao meu pai, Pedro, que sempre me ensinou a importância da perseverança e dedicação aos estudos, me motivando todos os dias. À minha mãe, Diana, por seu apoio inabalável.

À minha avó, Carmen, por sua sabedoria e por ter me mostrado que o conhecimento é a chave para o sucesso. Ao meu primo, Felipe, que tenho grande apreço, espero que este trabalho possa inspirá-lo a perseguir seus próprios sonhos e objetivos.

À minha amada namorada, Dayana, cujo apoio e incentivo foram inestimáveis em minha jornada. Espero que este trabalho possa ser uma fonte de orgulho e inspiração para você, assim como você é para mim.

Também gostaria de expressar minha gratidão aos professores do Instituto Metrópole Digital da UFRN, que me guiaram e inspiraram ao longo deste processo.

Aos Coordenadores do Programa de Residência em TI -TRF5, Prof. Dr Elias Jacob e Prof. Dr. Everton Cavalcante.

Ao meu Coorientador, David Montalvão, que me passou calma nos momentos que pensei que o trabalho não iria ser concluído e por me ajudar a adquirir dados que foram de difícil acesso.

Ao meu orientador neste trabalho, Prof. Dr. Daniel Sabino, por compartilhar seu vasto conhecimento e orientação ao longo deste trabalho. Sua ajuda e conselhos foram cruciais para o desenvolvimento de um trabalho de qualidade e aprendizado significativo.

*“A inteligência artificial pode descobrir soluções para problemas do passado, que mudarão a  
nossa forma de resolver os mesmos problemas no futuro.”  
(Dawiny Bastos)*

## Resumo

A tecnologia está cada vez mais difundida nos mais diversos ambientes profissionais. Não difere no ambiente jurídico, onde está cada vez mais buscando maneiras para auxiliar nas suas tarefas, visando melhorar seu desempenho, ajudando a sociedade a ter uma justiça de qualidade e menos morosidade. Um exemplo recente disso foi o sistema de Processo Judicial eletrônico(PJe), sendo um sistema de tramitação de processos judiciais cujo objetivo é atender as necessidades de diversos segmentos do Poder Judiciário brasileiro. Apesar de ter proporcionado melhorias no fluxo de processos, o PJe ainda apresenta limitações em relação às atividades repetitivas que consomem tempo dos servidores do Judiciário. A sobrecarga de processos judiciais excede a capacidade de recursos disponíveis, levando a uma justiça lenta e ineficiente. A necessidade de soluções que aumentem a agilidade no trâmite de processos é urgente, visando reduzir o congestionamento nos tribunais e melhorar a satisfação da sociedade com a justiça brasileira. Com o objetivo de aprimorar o desempenho do Judiciário brasileiro, este trabalho tem como meta desenvolver um modelo de *machine learning* utilizando técnicas de Processamento de Linguagem Natural (NLP) para priorizar processos judiciais com base no texto de suas petições iniciais. A partir da classificação em processos normais, preferenciais e urgentes, cada caso poderá ter um tratamento diferenciado, com priorização na ordem de julgamento. Para alcançar esse objetivo, adotamos uma estratégia de classificação em cascata, que começa pela classificação da petição inicial em normal ou preferencial. Se a petição for classificada como preferencial, verificamos se ela tem um teor mais urgente ou não. Os resultados dessa abordagem foram muito positivos. Obtivemos uma taxa de acerto superior a 90% para ambas as etapas de classificação. Isso significa que conseguimos identificar com precisão a maioria dos casos que exigem tratamento preferencial ou urgente, garantindo que esses processos sejam julgados com a prioridade que merecem. Espera-se que essa abordagem contribua para agilizar a tramitação dos processos e, consequentemente, reduzir a sobrecarga dos tribunais.

*Palavras-chave:* Machine Learning. Processamento de Linguagem Natural. Petição Inicial.



## Abstract

Technology is increasingly widespread in the most diverse professional environments. It is no different in the legal environment, where it is increasingly seeking ways to assist in its tasks, aiming to improve its performance, helping society to have a quality justice and less delay. A recent example of this was the Electronic Judicial Process (PJe) system, which is a system for processing judicial proceedings whose objective is to meet the needs of various segments of the Brazilian Judiciary. Although it has improved the flow of cases, PJe still has limitations in relation to repetitive activities that consume the time of the Judiciary's employees. The overload of judicial processes exceeds the capacity of available resources, leading to a slow and inefficient justice system. The need for solutions that increase the agility in the processing of cases is urgent, aiming to reduce congestion in the courts and improve society's satisfaction with Brazilian justice. Aiming to improve the performance of the Brazilian Judiciary, this paper aims to develop a machine learning model using Natural Language Processing (NLP) techniques to prioritize lawsuits based on the text of their initial petitions. From the classification in normal, preferential and urgent cases, each case may have a differentiated treatment, with prioritization in the trial order. To achieve this goal, we adopted a cascading classification strategy, which begins by classifying the text of the document as normal or preferential. If the text is classified as preferential, we verify whether it has a more urgent content or not. The results of this approach were very positive. We achieved a hit rate of more than 90% for both classification steps. This means that we were able to accurately identify most of the cases that require preferential or urgent treatment, ensuring that these cases are judged with the priority they deserve. It is hoped that this approach will contribute to speeding up the processing of cases and, consequently, reduce the burden on the courts.

*Keywords:* Machine Learning. Natural Language Processing. Initial Petition.

# Lista de ilustrações

Figura 1 – Inteligência artificial e seus subcampos	16
Figura 2 – Processamento de Linguagem Natural, Inteligência Artificial e Linguística	19
Figura 3 – Ilustração de n-grams	20
Figura 4 – Matriz de confusão genérica	22
Figura 5 – Diagrama de Fluxo do projeto	25
Figura 6 – Tabela de preferência legal do painel BI	26
Figura 7 – Bloxplot da quantidade de linhas	31
Figura 8 – Bloxplot da quantidade de caracteres	31
Figura 9 – Histograma da quantidade de linha	32
Figura 10 – Histograma da quantidade de caracteres	32
Figura 11 – Documento analisado aleatoriamente	33
Figura 12 – Proporção das petições normais e preferenciais	36
Figura 13 – Proporção das petições preferenciais e urgentes	40

# Lista de tabelas

Tabela 1 – Lista de atributos dos dados do TRF5	27
Tabela 2 – Parametros do TfidfVectorizer	38
Tabela 3 – Resultados para triagem de processos Normais e Preferenciais	41
Tabela 4 – Resultados para triagem de processos Preferenciais e Urgentes	42

# Lista de abreviaturas e siglas

IMD	Instituto Metrópole Digital
UFRN	Universidade Federal do Rio Grande do Norte
TRF5	Tribunal Regional Federal da 5ª Região
JFRN	Justiça Federal do Rio Grande do Norte
CNJ	Conselho Nacional de Justiça
PJe	Processo Judicial Eletrônico
SEEU	Sistema Eletrônico de Execução Unificada
IA	Inteligencia Artificial
SJ	Seção Judiciária
PLN	Processamento de Linguagem Natural
BI	Business Intelligence
TF-IDF	Term Frequency - Inverse Document Frequency
NaN	Not a Number

# Sumário

<b>1 INTRODUÇÃO</b>	<b>13</b>
1.1 Objetivos	14
1.2 Organização do trabalho	14
<b>2 Fundamentação teórica</b>	<b>15</b>
2.1 Inteligencia artificial	15
2.2 Machine learning	16
2.3 Processamento de linguagem natural	18
2.4 Métricas de avaliação	21
<b>3 METODOLOGIA DO TRABALHO E CONSIDERAÇÕES DE PROJETO</b>	<b>24</b>
3.1 Aquisição dos dados	25
<b>4 Desenvolvimento do projeto</b>	<b>27</b>
4.1 Análise exploratória dos dados	27
4.1.1 Dados provenientes do TRF5	27
4.1.2 Documentos PDF	30
4.2 Limpeza dos dados	33
4.3 Cruzamento dos dados	34
4.4 Pré-processamento do texto	35
4.5 Treinamento dos modelos	35
4.5.1 Triagem de processos normais e preferenciais	35
4.5.1.1 Separação dos dados	36
4.5.1.2 Treinamento	37
4.5.2 Triagem de processos preferenciais e urgentes	39
4.5.2.1 Separação dos dados	39
4.5.1.2 Treinamento	40
4.6 Resultados	41
<b>5 Considerações Finais</b>	<b>43</b>
5.1 Principais contribuições	43
5.2 Limitações	44
5.3 Trabalhos futuros	44
REFERÊNCIAS	46

# 1 Introdução

Segundo o Justiça em Números 2022, o Judiciário julgou 26,9 milhões de processos em 2021, o que significou uma alta de 11,1% em relação aos números de casos solucionados do ano anterior [1]. Um ótimo dado de fato, porém, ainda de acordo com o mesmo relatório, foi registrado 27,7 milhões de novas ações naquele mesmo período, o que representa 10,4% de crescimento. Atualmente, seria necessário um investimento substancial em recursos humanos para manter a estabilidade do fluxo de entradas e saídas de processos dentro do sistema de justiça. Entretanto, a justiça brasileira tem sido alvo frequente de críticas em relação aos seus altos custos. Na verdade, o orçamento do Poder Judiciário brasileiro é um dos maiores do mundo, correspondendo a uma parcela significativa do orçamento total do país.

Para enfrentar esses problemas, várias iniciativas foram lançadas nos últimos anos. Uma delas é a adoção de tecnologias e processos mais eficientes, como o uso de inteligência artificial (IA) e automação de processos. Hoje, com a ajuda do sistema de Processo Judicial eletrônico (PJe) [2], é possível automatizar alguns processos que levavam bastante tempo em outrora, além de deixar a disposição dados estruturados que podemos utilizar para devidas análises e eventualmente desenvolver inteligências que possibilite uma semi-automatização de um processo repetitivo que demanda grandes esforços.

O desenvolvimento de IA no âmbito jurídico está sendo cada vez mais difundido, inclusive, está presente na maioria dos tribunais brasileiros. Um levantamento realizado pelo Conselho Nacional de Justiça (CNJ) divulgou que até o meio do ano de 2022 existiam 111 projetos sendo desenvolvidos no Poder Judiciário que envolviam IA, o que representa um aumento de 171% em relação ao mesmo período do ano anterior, quando foram informados 41 projetos [3]. Alguns dos projetos inclusive já estão aptos a serem utilizados. Essas iniciativas são reguladas pelo CNJ, pela resolução n. 332/2020 [4], que instituiu o Sinapses (solução computacional, visando armazenar, testar, treinar, distribuir e auditar modelos de Inteligência Artificial) como a plataforma nacional de armazenamento, controle e versionamento dos modelos de IA, estabelecendo parâmetros de sua implementação e funcionamento.

Este trabalho visa ajudar o sistema jurídico a enfrentar o desafio de lidar com um excesso de processos que necessitam de um tratamento diferenciado, mas tem dificuldades de fazer isso com rapidez por falta de recursos humanos suficientes, apesar dos altos gastos do poder judiciário. A inovação proposta busca auxiliar os servidores a lidar com a demanda excessiva, o que pode levar a uma justiça mais ágil e eficiente, além de reduzir o congestionamento nos tribunais. As consequências dessa inovação podem ter um impacto significativo nas atividades diárias dos

servidores e magistrados, bem como na sociedade em geral. Os processos podem avançar mais rapidamente, proporcionando agilidade às necessidades dos envolvidos, especialmente aqueles que requerem urgência.

## 1.1 Objetivos

O objetivo deste trabalho é utilizar modelos de *Machine Learning* para realizar uma triagem de processos com diferentes graus de prioridade. Com a utilização desses modelos, espera-se otimizar o processo de triagem, evitando a necessidade do servidor responsável ler manualmente o texto da petição inicial para decidir o encaminhamento correto, o que resultará em agilidade na tramitação dos processos e, conseqüentemente, em melhorias na eficiência do sistema jurídico.

Os modelos desenvolvidos são independentes entre si e usarão o texto da petição inicial de cada processo como entrada para realizar as previsões de prioridade. Esse trabalho apresentará os seguintes passos para o desenvolvimento dos modelos mencionados:

- Descrição da origem dos dados utilizados;
- Realização de análise exploratória de dados para mapear as estratégias utilizadas na previsão do modelo;
- Separação dos dados em conjuntos de treino e testes;
- Análise de métricas de classificação;
- Escolha dos modelos.

Com estes passos, pretende-se desenvolver modelos precisos e confiáveis para uma boa estratégia de triagem dos processos, e, assim, contribuir para a eficiência do sistema jurídico.

## 1.2 Organização do trabalho

Este trabalho está estruturado em capítulos, organizados conforme o conteúdo e a ordem apresentada nos tópicos a seguir:

- O Capítulo 2 apresenta fundamentação teórica necessária ao entendimento do trabalho;
- O Capítulo 3 é detalhada a metodologia do trabalho, relatando a aquisição dos dados, e mostrando uma ideia de como será realizado o projeto;
- O Capítulo 4 apresenta a execução do projeto, desde da análise exploratória e limpeza dos dados até o treinamento e avaliação do modelo;
- Por fim, o Capítulo 5 apresenta as considerações finais, indicando suas limitações e possíveis melhorias.

## 2 Fundamentação teórica

Neste capítulo, será apresentada a fundamentação teórica do projeto, onde serão discutidos conceitos importantes das áreas de inteligência artificial, machine learning, processamento de linguagem natural, e métricas para avaliar modelos. Essa seção é fundamental para o entendimento dos conceitos básicos necessários para a execução do projeto, bem como para fornecer uma visão geral dos principais tópicos e teorias relacionadas à inteligência artificial e suas aplicações.

A compreensão desses conceitos é essencial para uma implementação bem-sucedida do projeto, permitindo que sejam feitas melhorias constantes nos modelos de aprendizado de máquina, a fim de se obter resultados mais precisos e confiáveis. Sendo assim, essa seção também servirá como uma referência valiosa para aqueles que desejam aprofundar seus conhecimentos em inteligência artificial e suas aplicações.

### 2.1 Inteligencia artificial

A definição de inteligência artificial é sugerida como “a capacidade de um computador digital ou robô controlado por computador para executar tarefas comumente associadas a seres inteligentes” [5] por BJ Copeland (Britannica, 2023). É um campo que está cada vez mais em expansão, e sendo aplicada em uma grande variedade de áreas e setores em todo o mundo, com o objetivo de melhorar a eficiência, a qualidade e a precisão de diversos processos e auxiliar as pessoas com demandas repetitivas.

Podemos destacar algumas aplicações [6]. Na saúde, por exemplo, a IA vem sendo utilizada para ajudar na identificação de doenças, na análise de imagens médicas e na análise de dados clínicos, permitindo um diagnóstico mais rápido e preciso para o paciente. Na indústria automotiva, temos exemplos de IA em sistemas de direção autônoma. No setor financeiro, temos exemplos aplicados em análise de riscos, previsões de mercado e detecção de fraudes. Na área de entretenimento, a IA é usada em jogos eletrônicos e em sistemas de recomendação em serviços de streaming, indicando conteúdos de acordo com as preferências do usuário.

Além disso, a IA vem sendo aplicada em diversos outros setores, como agricultura, varejo, logística, educação, entre outros. Em resumo, a IA mostra um enorme potencial para transformar diversos setores e áreas, trazendo melhorias significativas em termos de eficiência e rapidez em resolver problemas.

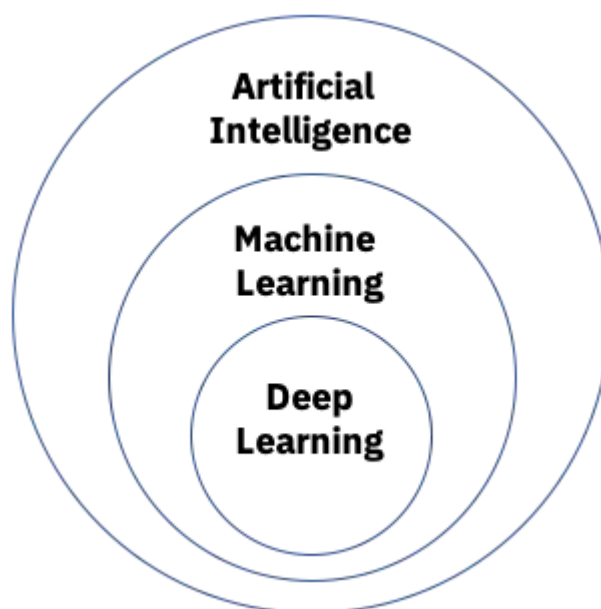
É importante mencionar que IA e *Machine Learning* são termos que estão em alta e são frequentemente confundidos, mas têm significados distintos. Enquanto a



IA se refere ao desenvolvimento de sistemas capazes de realizar tarefas que exigem inteligência humana, o *Machine Learning* é uma abordagem específica dentro da IA que permite que os sistemas aprendam a partir de dados, utilizando algoritmos e modelos estatísticos para identificar padrões.

Outro termo bastante falado no ecossistema da IA, é o de *Deep Learning*, que é um subcampo mais avançado do *Machine Learning*, mais complexo e que utiliza redes neurais artificiais com múltiplas camadas para aprender a partir dos dados [7]. A imagem abaixo ilustra o campo geral da inteligência artificial, bem como os subcampos do *Machine Learning* e *Deep Learning*, mencionados anteriormente.

Figura 1 – Inteligência artificial e seus subcampos.



Fonte: IBM Cloud Education(2020)

## 2.2 Machine learning

*Machine Learning* é “uma disciplina da área da Inteligência Artificial que, por meio de algoritmos, dá aos computadores a capacidade de identificar padrões em dados massivos e fazer previsões (análise preditiva). Essa aprendizagem permite que os computadores efetuem tarefas específicas de forma autônoma, ou seja, sem necessidade de serem programados” [8].

Os algoritmos de *Machine Learning* pode ser dividido em 4 categorias [9]:

- **Aprendizado supervisionado:** esses algoritmos usam um conjunto de dados rotulados para aprender padrões a partir de variáveis independentes e tomar decisões ou fazer previsões. Por exemplo, as informações coletadas do cliente de um banco, tais como idade, renda e histórico de inadimplência, são

variáveis independentes que podem ser utilizadas para avaliar a probabilidade de o cliente ser um bom pagador de empréstimo. Com o uso de algoritmos de aprendizagem supervisionada, é possível treinar um modelo que possa identificar padrões nesses dados e, assim, fazer previsões precisas sobre a probabilidade de inadimplência do cliente. Dessa forma, os bancos podem usar essa tecnologia para tomar decisões mais informadas sobre a concessão de empréstimos, minimizando os riscos de inadimplência e aumentando sua lucratividade.

- **Aprendizado não supervisionado:** esses algoritmos exploram dados sem rótulos para descobrir padrões que possam ajudar na organização dos dados. Por exemplo, podemos utilizar o histórico de compras, comportamento de navegação e os dados do usuário como idade, gênero, localização geográfica, entre outros e identificar um padrão de compras e agrupar usuários de um e-commerce e tirar proveito disso em direcionar melhor suas campanhas publicitárias, personalizar recomendações para determinado grupo de clientes, entre outras coisas.
- **Aprendizado semi-supervisionado:** nesse tipo de algoritmo, o sistema consegue lidar com dados rotulados ou não-rotulados. Geralmente, essa modalidade é empregada quando o custo para rotular os dados é muito elevado.
- **Aprendizado por reforço:** esse algoritmo permite que um sistema tome decisões com base em experiências anteriores, aprendendo com acertos e erros e sendo recompensado pelas decisões corretas. Esse algoritmo vem sendo aplicado em diversas áreas, incluindo a saúde, onde pode ajudar a melhorar a precisão dos diagnósticos. Ao utilizar algoritmos de aprendizado por reforço na área da saúde, é possível fornecer ao sistema dados de pacientes e sintomas, para o algoritmo poder aprender a tomar decisões mais precisas com base em sua avaliação em cada tentativa.

Um aspecto importante para ter sucesso no modelo de *Machine Learning*, é se preocupar desde a preparação dos dados e seu pré-processamento, até a sua avaliação. Antes de treinar um modelo, é importante que os dados estejam limpos, organizados e no formato adequado antes de usá-los na análise de negócios [10]. A preparação dos dados inclui várias etapas, como:

- **Coleta de dados:** O primeiro passo é coletar os dados de fontes relevantes e garantir que eles sejam completos e precisos.
- **Exploração dos dados:** determinar a qualidade dos dados, examinar sua distribuição e analisar a relação entre cada variável para entender melhor como compor uma análise.
- **Limpeza dos dados:** A limpeza de dados envolve a remoção de dados duplicados ou inconsistentes, bem como o tratamento adequado aos valores faltantes.

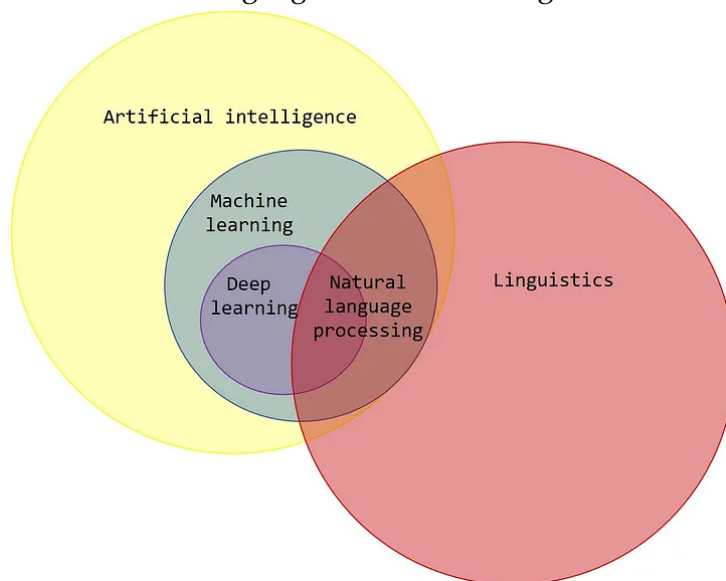
- Normalização dos dados: A normalização de dados envolve a transformação dos dados para ficarem todos em uma mesma escala. Isso ajuda a garantir que todos os dados sejam comparáveis e possam ser processados pelo modelo.
- Transformação dos dados: A transformação de dados envolve a conversão dos dados brutos em um formato adequado para o modelo. Isso pode incluir a criação de novas variáveis ou a redução da dimensionalidade dos dados.
- Separação dos dados: Os dados precisam ser divididos em conjuntos de treinamento, validação e teste. O conjunto de treinamento é usado para treinar o modelo, o conjunto de validação é usado para ajustar os hiperparâmetros do modelo e o conjunto de teste é usado para avaliar o desempenho final do modelo.

A preparação e pré-processamento de dados é uma etapa crítica em qualquer modelo de *machine learning* e a parte do projeto que leva mais tempo para ser concluída. No entanto, o esforço é recompensado com modelos mais precisos e confiáveis que podem ser usados para tomar decisões e prever resultados com maior precisão.

## 2.3 Processamento de linguagem natural

O Processamento de Linguagem Natural (PLN ou NLP) é uma “mescla entre ciência da computação, inteligência artificial e linguística, se dedicando a geração e compreensão automática da linguagem natural” [11]. Outra definição bem interessante é que “NLP é uma subárea de aprendizado de máquina que trabalha com linguagem natural, seja em texto ou áudio” [12]. Basicamente consiste em permitir que as máquinas entendam e produzam uma linguagem natural de forma semelhante a um ser humano.

Figura 2 – Processamento de Linguagem Natural, Inteligência Artificial e Linguística.



Fonte: Bruno Carloto (2021)

O PLN tem aplicações em diversas áreas, como a de tradução de idiomas, que permite aos tradutores obter resultados gramaticalmente corretos e precisos. Além disso, o PLN é utilizado em filtros de e-mail para detectar palavras, frases e padrões que sinalizam mensagens de spam, melhorando a segurança e a organização da caixa de entrada do usuário. Atualmente, serviços de e-mail como o Gmail usa o PLN para categorizar grupos de e-mails como “principal”, “social” ou “promoções”, facilitando a visualização dos e-mails importantes. Outra aplicação popular do PLN são as assistentes virtuais, como a Siri da Apple e a Alexa da Amazon, que utilizam o reconhecimento por voz para identificar padrões na fala e interpretar as solicitações dos usuários, executando tarefas específicas com base nas informações fornecidas [13].

O PLN também é utilizado na análise de sentimentos em redes sociais, classificação de documentos, chatbots e muitas outras aplicações. A evolução contínua do PLN tem ampliado suas possibilidades e contribuído para o desenvolvimento de soluções mais eficientes e personalizadas para atender às necessidades dos usuários.

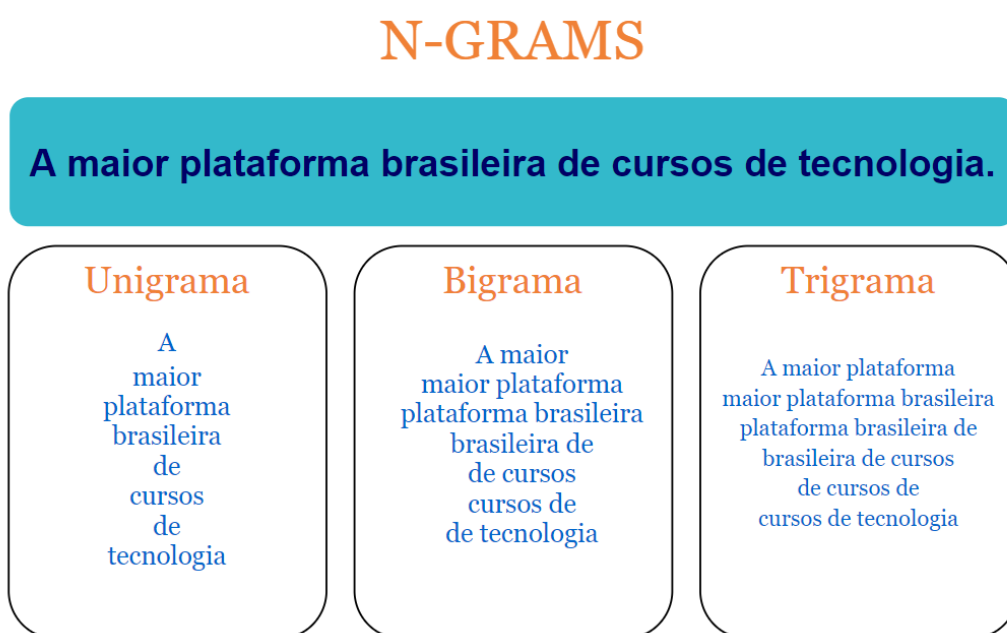
Existem diversas técnicas de PLN que podem ser aplicadas em uma variedade de situações para que ela seja compreendida pelas máquinas. Entre elas, temos o uso de um *corpus* para treinamento do algoritmo, a *tokenização* para divisão do texto em unidades menores, a normalização para deixar todas as palavras em um padrão comum, técnicas para retirar ruídos, como *stopwords* e pontuação, além do uso de *stemming* e *lemmatization* para reduzir as palavras à sua forma base. É importante

destacar que muitos desses termos estão interligados e são fundamentais para um bom processamento de linguagem natural [14].

Vamos conhecer os conceitos básicos e algumas técnicas, visto que muitos desses termos são interligados.

- **Corpus:** A palavra "corpus", originada do latim que significa "corpo", é utilizada para se referir ao corpo de um texto, que pode ser escrito ou falado, contendo um ou mais idiomas. Para representar uma coleção de textos, usamos o termo "corpora", que é o plural de "corpus". Essa coleção de textos pode ter um tema específico ou temas gerais.
- **Tokenização:** é uma etapa fundamental do processamento de linguagem natural, que consiste em dividir um texto em unidades menores, chamadas de tokens. Esses tokens podem ser palavras, sílabas, números, emojis, entre outros elementos que compõem o *corpus*. É importante destacar que os tokens podem ser compostos por mais de uma palavra, como palavras compostas.
- **Normalização:** é uma etapa importante do processamento de linguagem natural que visa tornar a análise mais precisa e consistente. Geralmente, ela é realizada após a *tokenização*. A normalização pode incluir a padronização de letras maiúsculas e minúsculas, bem como a combinação de frases semelhantes, para facilitar a análise, se desejado.
- **n-grams:** combinações de palavras que ocorrem juntas, normalmente com uma certa frequência

Figura 3 – Ilustração de n-grams.



- Pré-processamento e limpeza dos dados: nesta etapa consiste em limpar o texto, removendo ruídos como pontuação e caracteres especiais, bem como palavras que não agregam valor à análise, tais como as *stopwords*.
- StopWords: é uma técnica utilizada para remover as palavras de alta frequência e baixa relevância, tais como conectivos (por exemplo, "que", "o", "a", "de", entre outros) que não trazem informações importantes para o modelo. É uma etapa importante no pré-processamento do texto, pois ajuda a limpar o *corpus* e a reduzir a dimensionalidade dos dados. A biblioteca NLTK (National Language Toolkit) é uma das ferramentas comumente usadas para realizar a remoção das stop words. Ela oferece recursos para identificar e remover essas palavras, levando em conta o idioma do *corpus* em questão.
- Stemming: técnica para reduzir a palavra em seu radical. Por exemplo, as palavras "gato", "gata", "gatos" e "gatas" reduziriam-se para "gat" [15].
- Lemmatização: técnica para reduzir a palavra em seu lema, que é a forma masculina no singular. Por exemplo, as palavras "gato", "gata", "gatos" e "gatas" são todas formas do mesmo lema: "gato". A vantagem dessa técnica, bem como a de stemização, é a redução de vocabulário e abstração de significado [16].
- Bag of Words: técnica comum de Feature Extraction usada para transformar textos em dados estruturados. Ela representa o texto através da ocorrência de cada palavra, sem levar em conta sua ordem ou estrutura no texto. É como se todas as palavras fossem colocadas dentro de um saco.
- TF-IDF: significa Term Frequency - Inverse Document Frequency ou Frequência do Termo - Inverso da Frequência no Documento, em tradução livre para o português. É uma técnica que mede a importância de uma palavra em um texto, levando em conta a frequência de ocorrência da palavra no documento e a frequência da palavra em todo o *corpus*. Quanto mais frequente a palavra no documento e menos frequente em todo o *corpus*, mais importante ela é considerada.

O processamento de linguagem natural é uma área que apresenta uma série de desafios e oferece diversas aplicações práticas no nosso cotidiano. Além disso, com o avanço contínuo da tecnologia, há um enorme potencial para transformar a maneira como interagimos com dispositivos eletrônicos, como celulares e computadores, e outras tecnologias, abrindo um mundo de possibilidades para melhorar a comunicação entre humanos e máquinas.

## 2.4 Métricas de avaliação

Nessa etapa, é feita a avaliação do desempenho do modelo, a fim de determinar se ele consegue solucionar o problema proposto de forma satisfatória ou não. "Ao desenvolver projetos de *Machine Learning* e *Data Science*, é crucial a

utilização de métricas apropriadas para cada problema. O valor delas reflete a qualidade de um modelo, portanto se forem mal escolhidas, será impossível avaliar se o modelo de fato está atendendo os requisitos necessários” [17].

Existem diversas métricas disponíveis para avaliar o desempenho de modelos de *Machine Learning*, sendo que cada métrica é mais adequada para um tipo de problema. Classificação e regressão são utilizados métricas distintas. Como o nosso objetivo é avaliar modelos de classificação, vamos focar especificamente nas métricas de avaliação utilizadas nesse tipo de problema.

- **Matriz de confusão:** A matriz de confusão é “uma tabela que permite extrair métricas que auxiliam na avaliação de modelos de *Machine Learning* para classificação” [18]. Contudo, esta matriz busca entender a relação entre acertos e erros que o modelo apresenta. Ela mostra a frequência com que as classes verdadeiras são classificadas corretamente (verdadeiros positivos e verdadeiros negativos) e as frequências com que as classes verdadeiras são classificadas incorretamente (falsos positivos e falsos negativos). Pode parecer um pouco complexo, mas partindo de uma avaliação binária, onde existe apenas duas classes, podemos resumir em quatro valores iniciais, sendo:
  - **Positivo Verdadeiro** (True Positive – TP) Previstos positivos e são realmente positivos;
  - **Falso Positivo** (False Positive – FP) Previsto positivo e, na verdade, é negativo;
  - **Negativo Verdadeiro** (True Negative – TN) Negativos previstos e são realmente negativos;
  - **Falso Negativo** (False Negative – FN) Negativos previstos e, na verdade, positivos.

Figura 4 – Matriz de confusão genérica.

		Valor Predito	
		Sim	Não
Real	Sim	Verdadeiro Positivo (TP)	Falso Negativo (FN)
	Não	Falso Positivo (FP)	Verdadeiro Negativo (TN)

Fonte: Diego Nogare(2020)

- **Acurácia:** mede a proporção de resultados verdadeiros para o total de casos. “Ela deve ser usada em dados com a mesma proporção de exemplos para cada classe, e quando as penalidades de acerto e erro para cada classe forem as

mesmas” [19]. Quando temos classes desbalanceadas, podemos ter a falsa sensação de uma boa acurácia. Por exemplo, considere um conjunto de dados de transações bancárias, em que apenas 1% das transações são fraudulentas. Se um modelo classificar todas as transações como normais, terá uma acurácia de 99%, que é muito alta. Entretanto, esse modelo não conseguirá detectar as transações fraudulentas, o que é o problema principal que queremos resolver. Por isso, é importante usar outras métricas além da acurácia para avaliar o desempenho do modelo.

$$Acurácia = \frac{Positivos Verdadeiros + Negativos Verdadeiros}{Positivos Verdadeiros + Negativos Verdadeiros + Falsos Positivos + Falsos Negativos} \quad (1)$$

- Recall: também conhecida como revocação, é definido como a proporção de exemplos pertencentes a uma determinada classe corretamente identificados pelo modelo como pertencentes a essa classe, em relação ao número total de exemplos que realmente pertencem a essa classe, independentemente de serem classificados em outra.

$$Recall = \frac{Positivos Verdadeiros}{Positivos Verdadeiros + Negativos Falsos} \quad (2)$$

- Precisão: “A precisão mede o quanto podemos confiar num modelo quando ele prevê que um exemplo pertence a uma determinada classe” [20], ou seja, a precisão mede a quantidade de exemplos que o modelo classifica corretamente como positivos (verdadeiros positivos) em relação ao total de exemplos que ele classifica como positivos (verdadeiros positivos e falsos positivos).

$$Precisão = \frac{Positivos Verdadeiros}{Positivos Verdadeiros + Falsos Positivos} \quad (3)$$

- F1 Score: O F1 Score é uma métrica que combina precisão e recall de maneira equilibrada. “F1 é a média harmônica de recall (revocação) e precisão” [21]. O F1-score é especialmente útil quando se deseja avaliar o desempenho do modelo em situações em que há uma assimetria na distribuição das classes, isto é, quando uma classe tem muito mais exemplos do que a outra. Nesses casos, a acurácia pode ser enganosa, já que o modelo pode simplesmente prever sempre a classe majoritária e ainda assim ter uma acurácia alta. Já o F1-score leva em conta tanto a precisão quanto o recall, considerando o impacto das previsões erradas nas duas classes.

$$f1\ score = 2 * \frac{Precisão * Recall}{precisão + Recall} \quad (4)$$

É importante lembrar que não há uma única métrica que seja a melhor para avaliar o desempenho de um modelo. Cada métrica tem suas próprias vantagens e desvantagens e deve ser escolhida de acordo com o problema específico que está sendo abordado. Além disso, outras métricas também podem ser utilizadas dependendo do contexto, como a curva ROC, por exemplo. No entanto, a



compreensão dessa métrica requer um nível mais avançado de conhecimento, o que está além do escopo deste trabalho. Portanto, não abordaremos a curva ROC aqui.

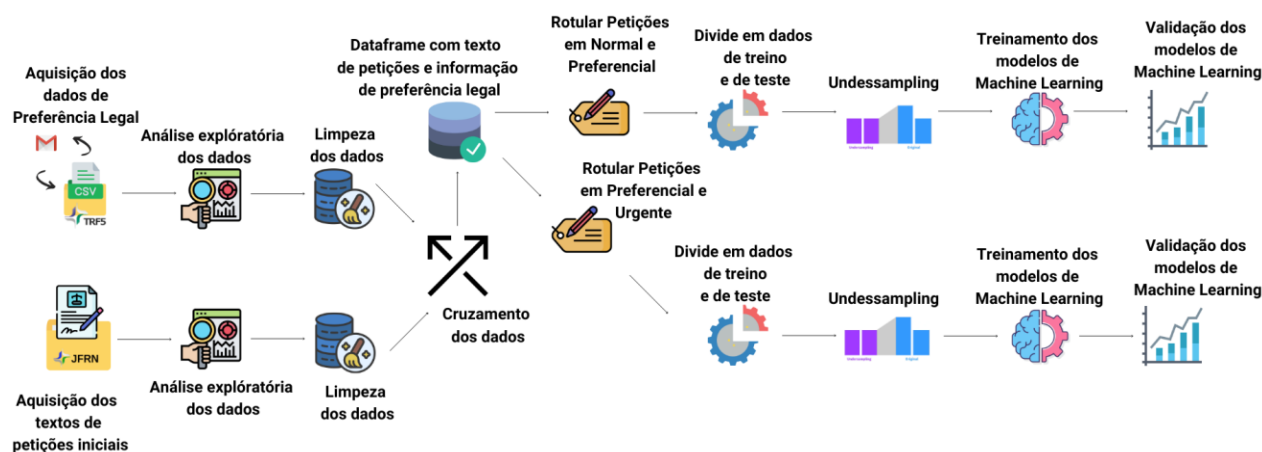
### 3 Metodologia do trabalho e considerações de projeto

A elaboração da monografia da residência em TI envolve a realização de um projeto, cujo resultado final é um produto entregável que deve ser útil e relevante para a instituição em que o residente está inserido. Para alcançar esse objetivo, é necessário um planejamento metódico e estratégico, que permita a utilização adequada de recursos e ferramentas para a concretização do projeto.

No caso da minha residência em TI no Tribunal Regional Federal da 5ª Região (TRF5), o projeto foi pensado em conjunto com pessoas de dentro da instituição, para atender às suas necessidades e expectativas. A partir das discussões e da coleta de informações, o projeto foi aprimorado e adaptado para o contexto específico do TRF5, de modo a proporcionar uma solução prática e efetiva.

Dentre as ferramentas utilizadas na construção do projeto, destaca-se a linguagem de programação *Python* [22], adotada por sua facilidade em trabalhar com dados e pela diversidade de bibliotecas disponíveis que facilitam o desenvolvimento de aplicações de alta qualidade. Essa linguagem é amplamente utilizada em aplicações de ciência de dados e *machine learning*, tornando-a uma escolha natural para a criação de soluções tecnológicas para esse projeto. O ambiente de desenvolvimento escolhido para a criação do projeto foi o *Google Colaboratory* [23]. Esta plataforma foi escolhida por sua capacidade de criar e executar códigos em um ambiente de computação em nuvem, o que oferece a conveniência de acessar o código a partir de qualquer máquina sem a necessidade de instalar softwares adicionais. Além disso, o uso da nuvem evita o consumo de recursos da máquina local. A figura 5 apresenta o diagrama de fluxo adotado no projeto, ilustrando cada etapa realizada.

Figura 5 – Diagrama de Fluxo do projeto.

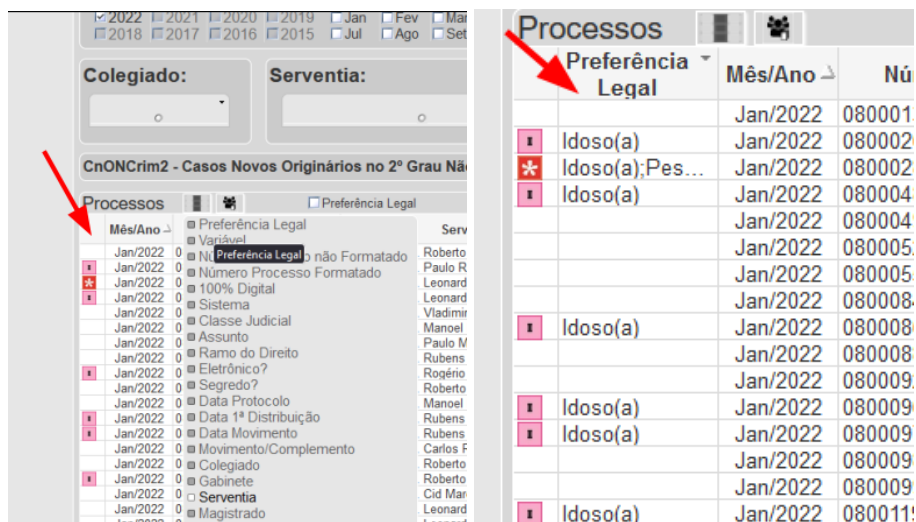


Fonte: Própria

### 3.1 Aquisição dos dados

A aquisição dos dados ocorreu em duas etapas. Na primeira, enviamos uma requisição de dados via e-mail ao Grupo de Trabalho do BI - TRF5, para obter informações de uma tabela que alimenta um painel de Business Intelligence do TRF5. Essa tabela contém informações importantes, como o número do processo e se o processo possui alguma preferência legal, bem como quais são essas preferências legais, porém, não inclui os textos das petições iniciais dos processos. Infelizmente, não foi possível adquiri-las através do TRF5. Abaixo, segue uma ilustração das tabelas dentro do painel BI. À esquerda, há uma coluna indicando se há preferência legal ou não no processo. Ao selecionar a opção "Preferência Legal", aparece a descrição correspondente, como ilustrado na imagem da direita.

Figura 6 – Tabela de preferência legal do painel BI.



Mês/Ano	Preferência Legal	Nú
Jan/2022		080001
Jan/2022	Idoso(a)	080002
Jan/2022	Idoso(a);Pes...	080002
Jan/2022	Idoso(a)	080004
Jan/2022		080004
Jan/2022		080005
Jan/2022		080005
Jan/2022		080008
Jan/2022	Idoso(a)	080008
Jan/2022		080008
Jan/2022		080009
Jan/2022	Idoso(a)	080009
Jan/2022	Idoso(a)	080009
Jan/2022		080009
Jan/2022		080009
Jan/2022	Idoso(a)	080011

Fonte: Própria

A informação sobre "Preferência Legal" é atribuída ao processo pelo advogado no momento do seu cadastro. No entanto, é importante ressaltar que o advogado pode, posteriormente, requerer a prioridade via petição, caso seja necessário. Nesse caso, cabe ao Servidor da Justiça realizar a marcação da prioridade solicitada. Esse processo de seleção é feito por meio de um campo específico no sistema, no qual o advogado pode escolher uma ou mais preferências legais que se aplicam ao caso em questão.

Na segunda etapa, foi utilizado documentos utilizados em uma monografia de um programa de residência anterior na instituição Justiça Federal do Rio Grande do Norte (JFRN). O trabalho do aluno consistiu em desenvolver modelos de *machine learning* para a designação de perícias em Juizados Especiais Federais. Durante o seu trabalho, ele teve acesso a uma base de dados de homologação do sistema Creta em um banco de dados Postgres, que continha o texto das petições iniciais em formato binário [24]. Ele transformou esses dados em texto e gerou cerca de 62 mil documentos em PDF. Consegui acesso a uma pasta já com os documentos em formato PDF, em momento algum tive acesso à base de dados, portanto, trata-se de um recorte de dados utilizados por outro residente, mas foi extremamente útil para o andamento desse trabalho.

## 4 Desenvolvimento do projeto

### 4.1 Análise exploratória dos dados

A análise exploratória de dados tem como objetivo investigar as principais características e padrões de um conjunto de dados por meio de técnicas gráficas e estatísticas descritivas. A partir dessa análise, podemos descobrir insights, identificar discrepâncias e anomalias nos dados, selecionar melhor nossos dados e ter uma eficiência maior no modelo de treinamento.

#### 4.1.1 Dados provenientes do TRF5

Foi recebido um arquivo no formato CSV que contém informações sobre os processos e sua preferência legal. O tamanho do arquivo é de 1,6 gigabytes, e ele contém um total de 10.971.997 registros organizados em 13 variáveis.

Tabela 1 – Lista de atributos dos dados do TRF5.

Atributo	Descrição
<i>%ID_PROCESSO_TRF</i>	As chaves de ligação são o resultado de transformações implementadas pela Governança do Tribunal Regional Federal da 5ª Região (TRF5) com o objetivo de padronizar a identificação dos processos. Essas chaves incluem informações como o sistema de origem do processo, a instância (em alguns sistemas) e a seção judiciária, seguidos pelo campo-chave que é utilizado para estabelecer a conexão entre os processos. Em outras palavras, as chaves de ligação são um mecanismo utilizado para garantir a consistência e a precisão na identificação dos processos dentro do TRF5
<i>Sistema</i>	Sistema de onde o processo se originou
<i>Data Protocolo</i>	Data em que o documento foi recebido e registrado no sistema de processos judiciais do tribunal
<i>Data Início Processo</i>	Data em que a petição inicial é protocolada no tribunal ou órgão jurisdicional competente

<i>Número Processo</i>	Código de identificação único que identifica o processo no sistema do tribunal ou órgão jurisdicional competente
<i>Eletrônico?</i>	Campo onde informa se é um processo eletrônico ou não
<i>Eletrônico/Físico</i>	Informa se é um processo eletrônico ou físico
<i>Instância Processo</i>	É a etapa em que se encontra o processo dentro da estrutura do poder judiciário. Cada processo passa por diferentes instâncias, que representam diferentes níveis de jurisdição
<i>Data Prim Distribuição</i>	Data em que o processo foi distribuído pela primeira vez para o juízo competente julgá-lo
<i>Data Últ Distribuição</i>	Data em que o processo foi redistribuído para outro juízo ou órgão competente após a sua tramitação inicial
<i>SJ Processo</i>	Seção Judiciária(SJ) é uma estrutura do Poder Judiciário responsável por garantir o acesso à justiça e a prestação jurisdicional em uma determinada região geográfica
<i>Flag Preferência Legal</i>	Campo que informa se aquele processo possui algum tipo de preferência legal ou não
<i>Preferência Legal</i>	Campo que informa qual ou quais preferencias legais o processo possui

Os dados obtidos do Tribunal Regional Federal da 5ª Região (TRF5) são provenientes de sistemas específicos utilizados pela instituição, incluindo o PJe (Processo Judicial Eletrônico), o Tebas (Sistema de Gestão de Processos Eletrônicos), o SEEU (Sistema Eletrônico de Execução Unificada), o PJeCNJ (versão do PJe adaptada para uso pelos Tribunais de Justiça de cada estado) e o Creta (Sistema de Controle de Recurso de Terceira Instância). Os dados obtidos foram coletados de todas as Seções Judiciárias que integram o Tribunal Regional Federal da 5ª Região (TRF5), abrangendo os estados de Alagoas, Ceará, Paraíba, Pernambuco, Rio Grande do Norte e Sergipe. Essas Seções Judiciárias são os tribunais federais de primeira instância que estão sob a jurisdição do TRF5 em cada um desses estados.

A base de dados obtida apresenta uma representação significativa dos processos eletrônicos em relação aos processos físicos. Cerca de 73% dos processos registrados na base já estavam no formato eletrônico, enquanto os outros 26% ainda se encontravam em formato físico. Vale mencionar que existem processos registrados na base que datam desde o ano de 1970, o que pode ter contribuído para a presença de uma quantidade significativa de processos físicos na amostra.

O campo 'Flag Preferência Legal' indica apenas se o processo tem alguma preferência legal ou não. Os registros que apresentam valor 'NaN' nesse campo correspondem a processos que não possuem nenhuma preferência legal, ou seja, estão sujeitos ao trâmite normal do sistema judiciário. Essa informação é relevante para a análise de prioridade de processos, pois permite identificar aqueles que possuem algum tipo de urgência ou relevância jurídica, como casos envolvendo idosos ou pessoas com deficiência, por exemplo.

O campo 'Preferência Legal' apresentava inicialmente mais de 40 categorias, mas durante a análise dos dados percebeu-se que em muitos casos eram apenas combinações de preferências. Para solucionar esse problema, realizou-se um procedimento de extração de classes únicas, resultando em apenas 11 classes totais:

- 'Deficiente Físico'
- 'Doença Terminal'
- 'Grande Devedor'
- 'Idoso(a)'
- 'Idoso(a) maior de 80 anos'
- 'Pessoa com Deficiência'
- 'Pessoa em Situação de Rua'
- 'Pessoas com deficiência (art. 9º, VII, da Lei nº 13.146/2015)'
- 'Preso/Acolhido/Internado'
- 'Prioridade de Tramitação'
- 'Réu Preso'

A combinação de preferências legais em um único processo pode ser explicada pelo fato de que, na hora de submeter a petição inicial, o advogado pode selecionar várias opções em um checkbox de múltipla escolha. Essas escolhas são registradas no banco de dados como uma lista separada por ';'. Por exemplo, se um advogado selecionar as preferências "Deficiente Físico" e "Idoso(a)", essas preferências serão registradas como "Deficiente Físico; Idoso(a)" no campo "Preferência Legal". Essa

categorização é importante para a análise de prioridades e urgências de cada processo.

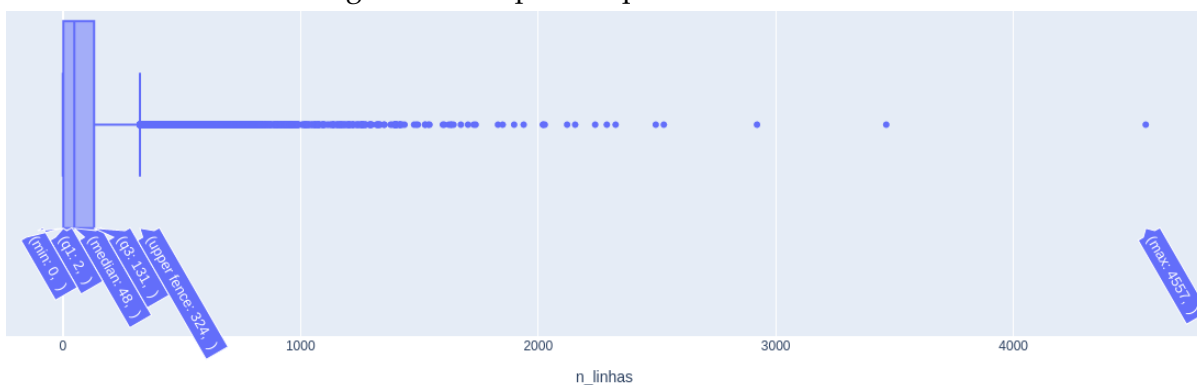
#### 4.1.2 Documentos PDF

Os documentos analisados são da Seção Judiciária do Rio Grande do Norte, fornecidos por ela. No entanto, é preciso destacar que alguns documentos foram erroneamente incluídos pelos advogados, requerendo uma análise para identificá-los e excluí-los. Os documentos em formato PDF foram submetidos ao processo de extração do texto contido neles e, posteriormente, incluídos em um *data frame* com apenas dois atributos: o conteúdo textual extraído do documento e o nome do arquivo. Vale ressaltar que o nome do arquivo segue um padrão específico, que consiste na pasta onde os documentos são armazenados, seguido do número do processo e finalizado com a extensão .pdf.

Com a finalidade de facilitar a análise dos textos extraídos, novos atributos foram criados. Foi observado que alguns documentos caracterizam-se por não possuírem linhas nem caracteres, e que, além disso, o primeiro quartil dos documentos contém apenas duas linhas ou menos, o que não é suficiente para caracterizar uma petição inicial. Esses casos podem ser considerados inserção de documentos errados pelo advogado no momento de abertura do processo. O objetivo é estabelecer um ponto de corte seguro de linhas e caracteres para filtrar documentos de petições iniciais, a fim de reduzir ao máximo a presença de documentos inseridos erroneamente.

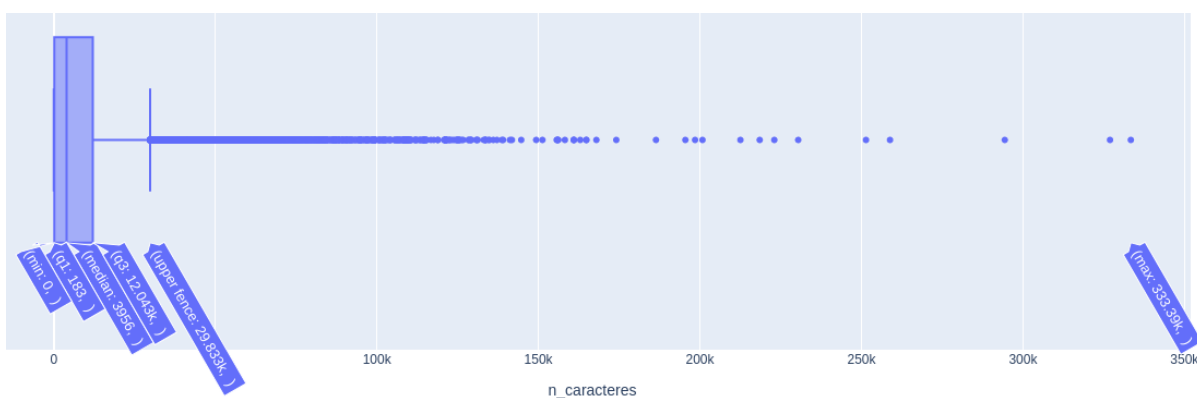
Analisando mais a fundo os dados, foram gerados boxplots para obter uma visão resumida da distribuição e valores discrepantes (outliers) do número de caracteres e linhas. As figuras 7 e 8 mostram que a maioria dos documentos contém entre zero e 30 mil caracteres e entre zero e 324 linhas. Com base nisso, foi possível traçar intervalos em um histograma e analisar melhor a distribuição de frequência do número de linhas e caracteres da maioria dos documentos fornecidos.

Figura 7 – Bloxplot da quantidade de linhas.



Fonte: Própria

Figura 8 – Bloxplot da quantidade de caracteres.

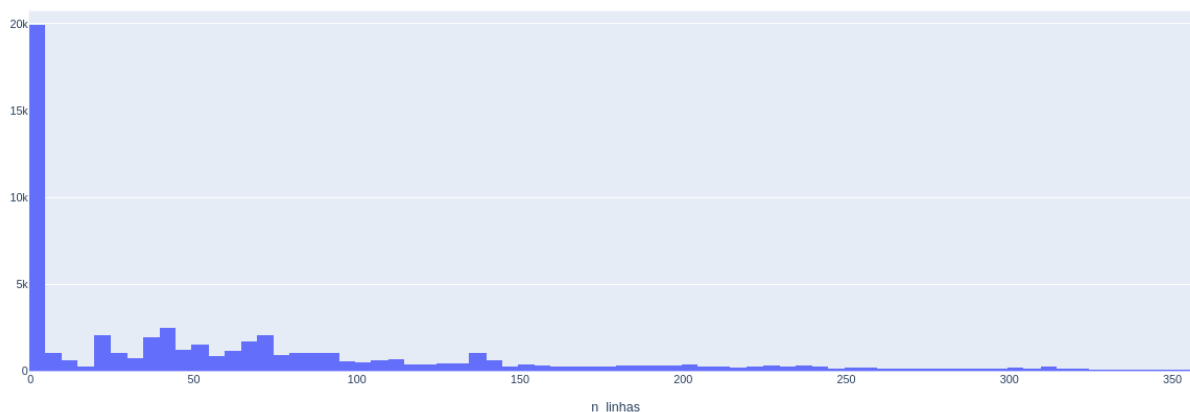


Fonte: Própria

Com o histograma plotado nas figuras 9 e 10, podemos observar que há uma proporção significativa de documentos com poucas linhas e caracteres. Por exemplo, muitos documentos têm menos de 4 linhas ou possuem menos de 500 caracteres. A análise desses dados sugere que é necessário investigar mais profundamente esses documentos, realizando a leitura manual desses documentos para constatar de fato que seja um documento errado inserido dentro do processo.

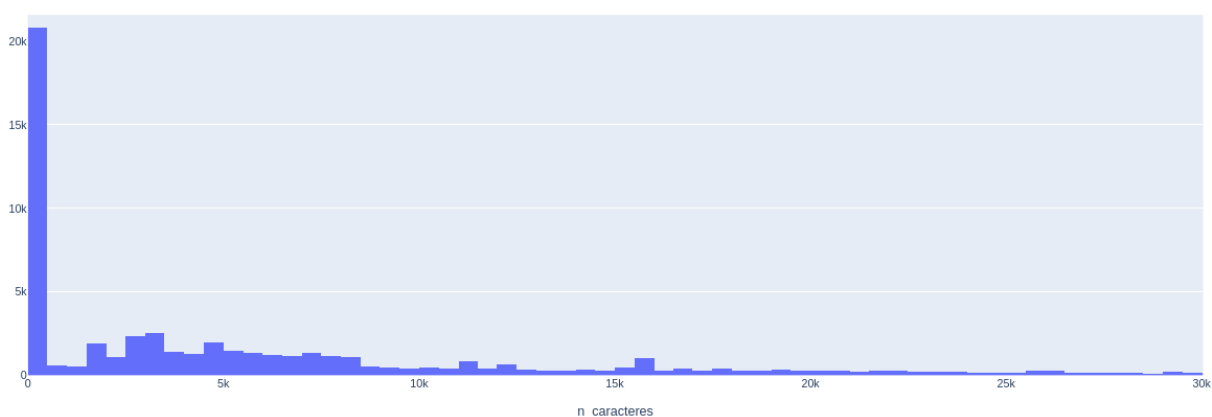


Figura 9 – Histograma da quantidade de linha.



Fonte: Própria

Figura 10 – Histograma da quantidade de caracteres.



Fonte: Própria

Após a análise inicial dos documentos, partimos para uma avaliação mais detalhada, realizando a leitura manual do conteúdo de uma amostra selecionada aleatoriamente. Rapidamente, foi percebido um padrão nos documentos com baixo número de linhas e caracteres. Esse padrão foi comprovado em diversos documentos iniciais, mostrando-se recorrente. A Figura 11 ilustra o conteúdo típico desses documentos com baixa quantidade de informações.

Figura 11 – Documento analisado aleatoriamente.



Fonte: Própria

Após identificar diversos documentos que indicavam se tratar apenas do Termo de Migração de Processo Físico, foi desenvolvida uma função para adicionar um novo atributo ao *data frame* indicando se o documento em questão é um termo de migração ou não. Aqui, foi utilizado uma função com expressão regular para identificar se havia a presença da frase “TERMO DE MIGRAÇÃO DE PROCESSO FÍSICO”.

Dos 62.200 documentos analisados, 3.197 eram esses termos. Essa descoberta reforça a hipótese de que existem documentos que não são petições iniciais inseridos no acervo de análise, o que ressalta a importância de uma limpeza criteriosa no *data frame* utilizado. Com isso, podemos reduzir ao máximo a presença de documentos que não são petições iniciais.

## 4.2 Limpeza dos dados

A limpeza dos dados é uma etapa crucial no processo de análise de dados, que envolve diversas atividades, tais como identificação e remoção de dados duplicados, tratamento de valores ausentes, correção de erros de digitação e verificação da integridade dos dados. Foi adicionada uma nova feature ao nosso *data frame*, contendo informações sobre o número do processo a partir do nome do documento. Após verificação, não foram encontrados processos duplicados na pasta.

Após identificar documentos inseridos erroneamente na pasta, é importante filtrar características relevantes para reduzir o número de documentos que não correspondem às petições iniciais, garantindo maior precisão e eficiência no treinamento do algoritmo. É importante lembrar que a inclusão acidental de documentos não relevantes na pasta de processo é um erro humano, podendo afetar negativamente a qualidade do nosso modelo.

Inicialmente, foram removidos todos os documentos que eram termos de migração de processo físico, já que não são relevantes para a análise de dados. Em seguida, foi estabelecido um critério para filtrar os textos com base no número de linhas, mantendo apenas os que possuíam entre 40 e 324 linhas e pelo menos 4000 caracteres, para garantir conteúdo relevante e eliminar possíveis outliers. O número máximo de linhas foi definido pela amplitude máxima do blox plot e os valores mínimos foram cuidadosamente testados para filtrar a maioria dos documentos indesejáveis. Vale destacar que haviam petições iniciais com menos de 40 linhas e com mais de 324 linhas. Optei por deixá-las de fora da análise para minimizar ao máximo a inclusão de documentos incorretos no conjunto de dados analisados.

É importante ressaltar que a limpeza de dados é um processo iterativo e pode ser aprimorado à medida que novos dados são adicionados e novas descobertas são feitas. Para pesquisas futuras, seria relevante consultar os servidores da justiça para identificar se os documentos com o maior número de linhas são de fato petições iniciais, assim podendo aumentar o número de documentos dentro do acervo analisado e enriquecer o trabalho.

### 4.3 Cruzamento dos dados

Nesta etapa, foi realizado cruzamento dos dados do *data frame* das petições iniciais com o *data frame* dos dados disponibilizados pelo TRF5. As variáveis-chave em ambos os conjuntos de dados são os números de processo, que atuam como identificadores únicos e, portanto, formam o ponto de conexão entre as duas bases de dados. Ao combinar esses conjuntos de dados, seremos capazes de criar um único *data frame* contendo informações do TRF5 e o texto da petição inicial correspondente, o que nos permitirá realizar o processamento de linguagem natural para criar nossos atributos e rotular as petições iniciais. Esses rótulos são importantes para treinar nosso modelo de aprendizado de máquina, o que nos permitirá classificar as petições iniciais de acordo com o que planejamos.

## 4.4 Pré-processamento do texto

Aqui, realizamos um processo de tratamento de texto com o objetivo de remover elementos que não sejam relevantes para a nossa análise. Para isso, removemos as “*stopwords*”, os espaços em branco, caracteres especiais e substituímos informações como números de telefone e e-mails por termos genéricos, como “*phonenumbr*” e “*emailaddress*”, respectivamente.

Além disso, para evitar possíveis vieses em relação à origem das petições, também retiramos o cabeçalho e rodapé dos documentos.

Por fim, é importante destacar que aplicamos a técnica de stematização para reduzir a forma das palavras em seu radical, o que permite contabilizar termos semelhantes como uma única ocorrência. Por exemplo, as palavras “juiz”, “juízes” e “juízas” serão consideradas como “juiz” após a stematização.

## 4.5 Treinamento dos modelos

Nessa etapa, realizaremos o treinamento dos nossos modelos de classificação. Dividiremos esse processo em duas partes: a primeira parte é responsável por identificar e separar os processos normais dos processos com alguma forma de preferência. Na segunda parte, utilizaremos outro modelo para classificar os processos preferenciais em dois grupos: aqueles com prioridade urgente e aqueles com prioridade padrão. Os processos urgentes terão prioridade ainda maior em relação aos processos preferenciais comuns.

O objetivo é que o modelo em produção faça a predição em cascata. Ou seja, se o processo for identificado com alguma preferência, ele passará por uma nova triagem para determinar se é um processo preferencial ou urgente. Dessa forma, garantimos que os processos com maior prioridade sejam atendidos de maneira ágil e eficiente, sem causar prejuízos para a parte autora.

Essa abordagem de classificação em cascata permitirá uma melhor precisão na classificação e garantirá que os processos urgentes sejam tratados com a devida prioridade.

### 4.5.1 Triagem de processos normais e preferenciais

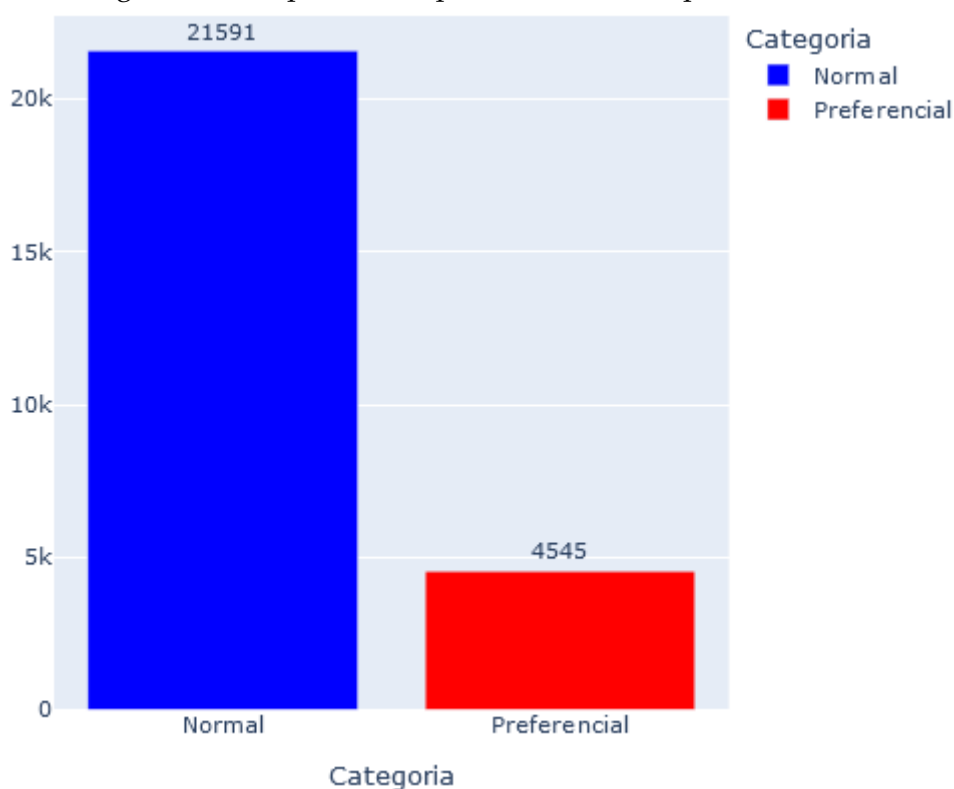
Primeiramente, será feita uma triagem para separar os processos em “preferenciais” e “não-preferenciais”. Os processos não-preferenciais seguirão o trâmite normal, sendo encaminhados para a secretaria da Vara. Já os que são

classificados como preferenciais passarão por uma nova triagem, sendo detalhada na próxima subseção.

#### 4.5.1.1 Separação dos dados

Para iniciar esta seção, é necessário rotular as petições iniciais em processos normais e preferenciais. Essa classificação foi realizada com base na feature "Flag Preferência Legal", a qual apresenta apenas duas opções: "None" e "Preferência Legal". O valor "None" indica que o processo não possui nenhuma preferência e, portanto, é considerado um processo normal. Já o valor "Preferência Legal" indica que o processo possui preferência e, conseqüentemente, é um processo preferencial. É importante salientar que a preferência legal pode ser determinada por diversos motivos, como idade avançada, doença grave, entre outros fatores que demandam uma atenção especial por parte do sistema judicial. A figura 12 nos mostra a quantidade de processos de cada classe a ser trabalhada nesse momento.

Figura 12 – Proporção das petições normais e preferenciais.



Fonte: Própria

O próximo passo para esta etapa é definir nossas variáveis independentes e nossa variável alvo. As variáveis independentes serão compostas pelas palavras

presentes nos textos, que serão pré-processadas com algumas técnicas de Processamento de Linguagem Natural.

Para garantir que nosso modelo de aprendizado de máquina aprenda de forma equilibrada as diferentes classes de nosso conjunto de dados, é importante separar nossos dados de treino e teste de maneira apropriada. Para isso, podemos usar a função `"train_test_split"` da biblioteca `"scikit-learn"`, uma das mais conhecidas para trabalhar com *Machine Learning*.

Ao dividir nossos dados em conjuntos de treino e teste, definimos um tamanho de teste de 20%, o que significa que 80% dos dados serão usados para treinar o modelo e os outros 20% para avaliar o quão bem ele foi treinado. No entanto, é importante notar que nosso conjunto de dados está desbalanceado, como já vimos na figura 11, o que pode levar o algoritmo de aprendizado de máquina a ter dificuldades em distinguir a classe minoritária devido à falta de amostras em comparação com a classe majoritária. Isso pode afetar negativamente a precisão e outras métricas adequadas do nosso modelo.

Para lidar com esse problema, podemos usar uma técnica chamada *"undersampling"*, que consiste em selecionar aleatoriamente algumas amostras da classe majoritária para igualar o número de amostras da classe minoritária [25]. Essa técnica pode ser facilmente implementada usando a classe `"RandomUnderSampler"` do pacote `"imbalanced-learn"`. Ao usar essa técnica, podemos garantir que nosso modelo seja treinado com uma quantidade igual de amostras de cada classe, o que ajuda a tornar o aprendizado mais equilibrado e justo. Após aplicada a técnica, utilizaremos 3636 documentos de cada classe para o treinamento do modelo.

É importante destacar que os dados balanceados com a técnica de *"undersampling"* serão utilizados apenas no momento do treino. Na validação, é necessário usar dados não balanceados. Isso porque, em um cenário real, a ocorrência de processos rotulados como "normal" é geralmente maior do que a de processos rotulados como "preferencial". Caso utilizemos dados balanceados para validar nosso modelo, estaríamos distorcendo a realidade. Portanto, é importante manter a integridade dos dados de validação e testá-los em sua forma original para obter uma avaliação mais precisa e confiável do modelo. Logo, teremos 4319 documentos da classe "normal" e 909 da classe "preferencial" para realizarmos o teste e uma avaliação que seja condizente com a realidade.

#### 4.5.1.2 Treinamento

Nesta etapa, vamos definir como iremos utilizar o Processamento de Linguagem Natural e configurar os parâmetros para o `TfidfVectorizer`, que é um módulo do pacote `"scikit-learn"`. Sua função é converter uma coleção de textos brutos em uma matriz de recursos *TF-IDF*, que mede a importância de cada palavra em

relação aos documentos da coleção. Optei por utilizar esse vetorizador por já conhecer as técnicas de *bag of words* e *TF-IDF*, e reconhecer que este último é uma abordagem mais eficiente.

Tabela 2 – Parâmetros do TfidfVectorizer.

Parâmetro	Valor	Definição
<i>max_features</i>	2000	Constrói um vocabulário que leva em consideração apenas as palavras mais frequentes em todo o <i>corpus</i> , ordenadas por frequência de termo, com um limite máximo definido pelo parâmetro
<i>ngram_range</i>	(1, 3)	O intervalo de valores n define o limite inferior e superior para os diferentes n-gramas que serão extraídos.

Nesta etapa, foram realizados testes com diferentes configurações para o *ngram\_range*. Foram testados modelos com apenas unigrama, apenas bigrama, apenas trigrama. Também foram testados os conjuntos de unigrama e bigrama, depois os bigrama e trigrama e em seguida, conjuntos de unigrama, bigrama e trigrama. O resultado mais promissor foi obtido com a utilização de unigrama, bigrama e trigrama. Quanto à quantidade de *features*, limitamos o número de palavras mais frequentes do *corpus* para reduzir o tempo de processamento de treino e teste dos modelos. Observamos que, ao ultrapassar 2000 *features*, não houve melhora significativa nas métricas dos modelos escolhidos, portanto, optamos por estabelecer o limite em 2000 *features*.

Para a etapa de treino, escolhemos utilizar três modelos diferentes: o *Random Forest*, o *Support Vector Machines* (SVC) e o Classificador de Árvores Extras (*ExtraTreesClassifier*). Esses modelos foram selecionados, por já serem conhecidos por mim, o que facilitou o trabalho. Iniciamos o processo com os valores padrões dos hiperparâmetros de cada modelo e, posteriormente, realizamos modificações manuais com o objetivo de encontrar melhorias e compará-las com o desempenho do modelo original sem as alterações. Os resultados obtidos serão apresentados na seção de resultados.

## 4.5.2 Triagem de processos preferenciais e urgentes

Os processos classificados na etapa anterior como preferenciais serão encaminhados diretamente para a assessoria e serão submetidos a uma nova triagem para classificá-los como "urgentes" ou não. Os processos não urgentes serão chamados apenas de processos "preferenciais". A grande diferença entre eles é que os processos classificados como "urgentes" não precisarão esperar por uma sentença para terem um ato jurisdicional.

### 4.5.2.1 Separação dos dados

Na etapa atual da análise, usaremos a feature "Preferência Legal" para classificar as petições iniciais em preferenciais e urgentes. Essa feature nos informa quais preferências legais estão anexadas a cada processo e, como descobrimos na análise exploratória dos dados, alguns processos possuem mais de uma preferência legal, e na base de dados estão separadas por ";".

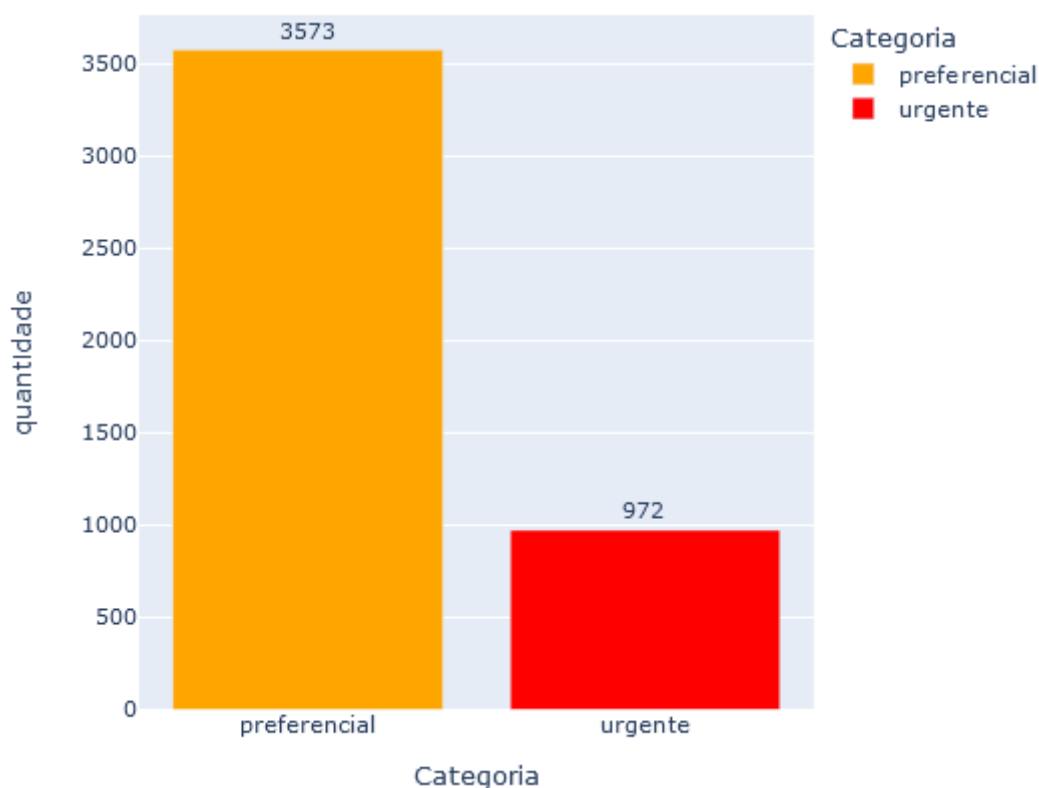
Para essa etapa, decidimos excluir os processos normais, uma vez que não são relevantes para a nossa análise. Nosso objetivo agora é focar nos processos preferenciais e urgentes. Para determinar quais preferências legais se encaixam em processos que precisam de urgência na tramitação, foi conversado com servidores da Justiça Federal do Rio Grande do Norte. A partir dessa conversa, descobriu-se que apenas as preferências legais "Doença Terminal", "Preso/Acolhido/Internado" e "Réu Preso" são consideradas processos urgentes.

Para lidar com essa informação, criamos uma nova feature chamada "preferencia\_urgencia". Através de um mapeamento com Python, e utilizando expressões regulares para identificar os termos de interesse, podemos identificar a existência dessas três preferências legais dentro do campo contendo as informações e classificar o processo como preferencial ou urgente, conforme a presença dessas preferências.

Dessa forma, com essa nova feature, podemos classificar as petições iniciais de acordo com sua urgência e prioridade, o que pode ajudar a otimizar a priorização dos processos. Na figura 13, podemos ver que temos um novo conjunto de dados desbalanceados. Para lidar com essa situação, precisamos repetir todas as etapas que fizemos no treinamento das classes normais e preferenciais.



Figura 13 – Proporção das petições preferenciais e urgentes.



Fonte: Própria

Utilizamos a técnica "*undersampling*" nesse novo conjunto de dados e, após essa etapa, ficamos com 778 processos de cada classe para treino. Para validação, escolhemos um conjunto de dados para simular um cenário real. Para isso, selecionamos 715 processos para a classe preferencial e 194 processos para a classe urgente.

#### 4.5.1.2 Treinamento

Durante o treinamento e validação dos processos preferenciais e urgentes, foi decidido manter os mesmos parâmetros utilizados na tabela 2 para o *TfidfVectorizer*. Esse modelo foi configurado para trabalhar com unigrama, bigrama e trigrama, selecionando apenas os 2000 conjuntos de *n-grams* mais frequentes dentro do *corpus*, da mesma forma que os modelos da etapa anterior.

No momento do treinamento, foram utilizados os modelos *Random Forest*, *Support Vector Machines (SVC)* e *ExtraTreesClassifier*, como na experiência anterior. Em

seguida, foram realizados testes com diversas alterações nos hiperparâmetros desses modelos, mas a maioria delas resultou em pouca melhoria na performance do modelo. Apesar de não termos encontrado uma combinação de hiperparâmetros que resultasse em uma grande melhoria na performance, já nos proporcionou um ótimo resultado. Agora vamos analisar os resultados obtidos na próxima seção.

## 4.6 Resultados

Podemos analisar os resultados do primeiro algoritmo, cujo objetivo é prever se a petição inicial é um processo normal ou um processo preferencial, por meio de diversas métricas presentes no relatório de classificação. A acurácia do nosso algoritmo apresentou um valor acima de 90% para todos os modelos testados. No entanto, se nos atentarmos apenas à acurácia, perderemos informações relevantes para a avaliação do modelo. Por isso, é importante também analisar o recall, que mede a proporção de exemplos que foram corretamente identificados pelo modelo, e a precisão, que mede a proporção de exemplos classificados naquela determinada classe, que são realmente daquela classe.

Durante nossos testes, verificamos que as métricas já apresentavam resultados satisfatórios com os parâmetros padrões de cada modelo. No entanto, realizamos alterações nos hiperparâmetros a fim de maximizar o desempenho dos modelos, o que resultou em um incremento médio de 1 a 2% no recall. Nas tabelas 3 e 4, é possível visualizar os resultados alcançados pelos nossos modelos em seu melhor desempenho. Diante da proposta do trabalho, é importante ressaltar que a métrica mais relevante vai ser o recall das classes que mais necessitam de celeridade no processo, ou seja, as classes preferencial e urgente.

Tabela 3 – Resultados para triagem de processos Normais e Preferenciais.

Métricas	Classe	RF	SVM	ExtraTree
<b>Acurácia</b>		91%	92%	92%
<b>Recall</b>	Normal	92%	92%	92%
	Preferencial	91%	90%	89%
<b>Precisão</b>	Normal	98%	98%	98%
	Preferencial	69%	70%	70%
<b>F1-Score</b>	Normal	95%	95%	95%
	Preferencial	79%	79%	79%

Embora os modelos de classificação apresentem uma alta taxa de acurácia (recall) em ambas as classes, com valores acima de 90% em todos eles, é preciso levar em conta a frequência desproporcional da classe "normal" em relação à classe "preferencial". Essa desproporção pode levar a um grande número de petições serem classificadas erroneamente como "preferenciais" quando comparado ao número real de petições preferenciais que chegam na justiça, prejudicando a métrica de precisão do nosso modelo, onde alcançamos 70%. Isso significa que 30% do total de processos que vão passar para a próxima etapa, na verdade, são processos sem preferência. Dos 3 modelos analisados, vale destacar o desempenho do *Random Forest*, onde seu recall foi superior aos demais na classe preferencial, que é a classe mais importante a ser classificada corretamente.

Já ao analisar o desempenho do segundo algoritmo, cujo objetivo é prever se uma petição inicial é um processo preferencial ou urgente, todos os modelos apresentaram uma acurácia de 94%, e as demais métricas foram similares, conforme a tabela 4, com uma ligeira vantagem do modelo SVM, tendo o melhor desempenho na classe urgente, que é a mais importante a ser classificada corretamente.

Tabela 4 – Resultados para triagem de processos Preferenciais e Urgentes.

Métricas	Classe	RF	SVM	ExtraTree
<b>Acurácia</b>		94%	94%	94%
<b>Recall</b>	Preferencial	95%	95%	95%
	Urgente	90%	91%	90%
<b>Precisão</b>	Preferencial	97%	98%	97%
	Urgente	83%	83%	83%
<b>F1-Score</b>	Preferencial	96%	96%	96%
	Urgente	86%	87%	87%

Em geral, esses resultados sugerem que os modelos são capazes de identificar corretamente processos preferenciais e urgentes, o que pode ser útil para agilizar e priorizar o fluxo de trabalho. Dessa forma, ao considerarmos tanto a precisão quanto o recall, podemos avaliar o desempenho do modelo de maneira mais abrangente, levando em conta as particularidades de cada classe. Essa análise mais completa dos resultados nos permite compreender melhor as limitações e oportunidades do modelo e, assim, realizar ajustes e melhorias caso necessário.

## 5 Considerações Finais

Neste trabalho, procurou-se desenvolver um projeto que pudesse ser relevante no contexto jurídico, contribuindo para acelerar os processos, especialmente aqueles que requerem atenção especial.

O objeto do estudo foi prever os modelos em cascata, em que primeiro faz a categorização dos processos normais e os processos que têm alguma forma de preferência. Em seguida, caso o processo tenha alguma preferência, faz a rotulagem em processos preferenciais e processos urgentes. Ambos os modelos apresentaram métricas relevantes, mas há espaço para melhorias.

É importante destacar que este projeto tem como objetivo ajudar ao Tribunal Regional Federal da 5ª Região a lidar com o grande volume de processos, facilitando a identificação daqueles que requerem maior atenção e agilizando seu andamento. Espera-se que, com a aplicação das técnicas de processamento de linguagem natural, seja possível tornar o processo mais eficiente e reduzir o tempo de espera dos envolvidos. Acredito que essa iniciativa possa ser replicada em outros tribunais, trazendo benefícios para todo o sistema judicial do país.

### 5.1 Principais contribuições

A principal contribuição deste trabalho foi dar início a um estudo criterioso feito com uma metodologia sólida com estudos de *machine learning* e processamento de linguagem natural para classificar textos de petições iniciais com seu determinado grau de preferência. A expectativa em cima do trabalho, quando realmente for colocado em produção, é ter uma contribuição muito grande a quem utiliza o judiciário brasileiro.

Com este estudo, era visado fornecer uma contribuição significativa para a justiça e para aqueles que utilizam seus serviços, tais como:

- Fazer com que os Servidores da Justiça economizem tempo de leitura feita para que aquele determinado processo seja encaminhado para o setor correto e tenha seu trâmite de acordo com sua necessidade, fazendo com que o tempo que sobra possa realizar outras atividades
- Dar opções de melhorias e contribuir com a transformação digital do Poder Judiciário
- Reduzir o tempo de espera da parte autora do processo quando é necessário uma certa urgência

Porém, caso o modelo fosse implementado hoje, e se suas métricas na vida real fossem semelhantes às métricas estudadas em nossos resultados, os usuários que precisam de urgência nos seus processos seriam os mais beneficiados, já que o modelo classifica corretamente mais de 90% desses casos, isso realmente iria trazer celeridade a esses processos. No entanto, é importante ressaltar que a precisão do modelo em identificar a urgência de um processo ainda precisa ser aprimorada, já que cerca de 17% dos processos classificados como urgentes pelo modelo não são urgentes, e 30% dos processos classificados com alguma preferência, não tem preferência. Isso significa que os Servidores da Justiça ainda precisarão ler o conteúdo da petição inicial em determinado momento para verificar se o processo foi classificado corretamente e enviá-lo para o local apropriado.

Com o bom resultado obtido, o presente estudo representa um importante passo para aprimorar a eficiência do sistema judiciário brasileiro, e pode ser um ponto de partida para futuras melhorias na identificação do grau de preferência dos processos.

## 5.2 Limitações

A principal limitação do trabalho reside na grande quantidade de processos normais que passam pela triagem junto aos processos preferenciais e urgentes. Embora o modelo apresente uma taxa de erro de apenas 8% ao classificar processos normais, considerando em um cenário real em que há um número muito maior de processos desse tipo em comparação com os preferenciais, muitos acabam sendo encaminhados para a próxima etapa da triagem, mesmo não possuindo preferências legais. Essa sobrecarga pode afetar a eficiência do processo de triagem, requerendo uma maior atenção do Servidor da Justiça que trabalha na Assessoria, tendo que retornar aquele processo à Secretaria da Vara responsável.

## 5.3 Trabalhos futuros

Esse trabalho apesar de apresentar bons resultados, sempre vai haver espaço para melhorias. Um dos trabalhos que pode ser realizado é a inclusão de novos dados, provenientes de todas as fontes possíveis, todas as seções judiciárias, para aprimorar ainda mais a classificação de processos, sobretudo, as classificadas como normais, sem que haja prejuízo na precisão da triagem dos processos preferenciais. Além disso, é importante explorar novos modelos de *machine learning*, inclusive, desenvolver um algoritmo de rede neural, que costuma ter um ótimo desempenho quando se trabalha com textos.

Um trabalho que será útil é a implementação da API do modelo, para permitir

que os servidores testem a entrada de dados e avaliem a eficiência do modelo em um cenário real. Ao disponibilizar o modelo para testes, será possível coletar feedback dos usuários e dos próprios servidores, permitindo uma avaliação mais completa do seu desempenho. Isso pode levar a ajustes e melhorias no modelo, tornando-o ainda mais eficiente em identificar o grau de preferência dos processos. Todas essas medidas podem garantir um processo de triagem ainda mais eficiente e preciso, o que resultaria em benefícios significativos para todos os envolvidos no sistema judiciário.

## Referências

[1] Justiça em Números 2022: Judiciário julgou 26,9 milhões de processos em 2021. Disponível em: <<https://www.cnj.jus.br/justica-em-numeros-2022-judiciario-julgou-269-milhoes-de-processos-em-2021/>>. Acesso em: 24 fev. 2023.

[2] PJe. Disponível em: <[https://www.pje.jus.br/wiki/index.php/P%C3%A1gina\\_principal](https://www.pje.jus.br/wiki/index.php/P%C3%A1gina_principal)>. Acesso em: 2 mar. 2023.

[3] Justiça 4.0: Inteligência Artificial está presente na maioria dos tribunais brasileiros. Disponível em: <<https://www.cnj.jus.br/justica-4-0-inteligencia-artificial-esta-presente-na-maioria-dos-tribunais-brasileiros/>>. Acesso em: 27 fev. 2023.

[4] Disponível em: <<https://atos.cnj.jus.br/atos/detalhar/3429>>. Acesso em: 27 fev. 2023.

[5] **Artificial intelligence | Definition, Examples, Types, Applications, Companies, & Facts | Britannica.** Disponível em: <<https://www.britannica.com/technology/artificial-intelligence>>. Acesso em: 28 fev. 2023.

[6] 10 usos de IA que estão transformando a saúde. Disponível em: <<https://uds.com.br/blog/10-usos-de-ia-que-estao-transformando-a-saude/>>. Acesso em: 28 fev. 2023.

[7] **Qual a diferença entre ia, machine learning e deep learning?** Disponível em: <<https://pt.linkedin.com/pulse/qual-diferen%C3%A7a-entre-ia-machine-learning-e-deep-garc%C3%AAs-de-castro>>. Acesso em: 1 mar. 2023.

[8] **Conheça os principais benefícios do Machine Learning.** Iberdrola. Disponível em: <<https://www.iberdrola.com/inovacao/o-que-e-machine-learning>>. Acesso em: 2 mar. 2023.

- [9] **Machine learning: o que é, para que serve + exemplos.** Disponível em: <<https://www.zendesk.com.br/blog/machine-learning/>>. Acesso em: 2 mar. 2023.
- [10] **O que é preparação de dados?** Disponível em: <<https://www.alteryx.com/pt-br/glossary/data-preparation>>. Acesso em: 2 mar. 2023.
- [11] **PLN: o que é Processamento de Linguagem Natural?** Alura. Disponível em: <<https://www.alura.com.br/artigos/o-que-e-pln>>. Acesso em: 3 mar. 2023.
- [12] CARLOTO, Bruno. Natural Language Processing Pipeline. Disponível em: <<https://brunosjob-analytics-notebook.medium.com/natural-language-processing-pipeline-c697e9defb59>>. Acesso em: 3 mar. 2023.
- [13] **Oito exemplos comuns de processamento de linguagem natural e seu impacto na comunicação.** Tableau. Disponível em: <<https://www.tableau.com/pt-br/learn/articles/natural-language-processing-examples>>. Acesso em: 3 mar. 2023.
- [14] **Guia de NLP - conceitos e técnicas.** Alura. Disponível em: <<https://www.alura.com.br/artigos/guia-nlp-conceitos-tecnicas>>. Acesso em: 3 mar. 2023.
- [15] **Lemmatization vs. stemming: quando usar cada uma?** Alura. Disponível em: <<https://www.alura.com.br/artigos/lemmatization-vs-stemming-quando-usar-cada-uma>>. Acesso em: 4 mar. 2023.
- [16] RODRIGUES, Jéssica. O que é o Processamento de Linguagem Natural? Disponível em: <<https://medium.com/botsbrasil/o-que-%C3%A9-o-processamento-de-linguagem-natural-49ece9371cff>>. Acesso em: 4 mar. 2023.
- [17] KUNUMI. Métricas de Avaliação em Machine Learning: Classificação. Disponível em: <<https://medium.com/kunumi/m%C3%A9tricas-de-avalia%C3%A7%C3%A3o-em-machine-learning-classifica%C3%A7%C3%A3o-49340dcdb198>>. Acesso em: 6 mar. 2023.
- [18] PRATES, Wlademir Ribeiro. O que é a Matriz de Confusão? Disponível em: <<https://cienciaenegocios.com/o-que-e-a-matriz-de-confusao/>>. Acesso em: 6 mar. 2023.



[19] FILHO, Mario. **As Métricas Mais Populares para Avaliar Modelos de Machine Learning**. Disponível em:

<<https://mariofilho.com/as-metricas-mais-populares-para-avaliar-modelos-de-machine-learning/>>. Acesso em: 6 mar. 2023.

[20] FILHO, Mario. **Precisão, Recall e F1 Score Em Machine Learning**. Disponível em: <<https://mariofilho.com/precisao-recall-e-f1-score-em-machine-learning/>>. Acesso em: 7 mar. 2023.

[21] HARRISON, M. **Machine Learning – Guia de Referência Rápida**. [s.l.] Novatec Editora, 2019.

[22] **Welcome to Python.org**. Python.org. Disponível em: <<https://www.python.org/>>. Acesso em: 19 mar. 2023.

[23] **Google Colaboratory**. Disponível em: <<https://colab.research.google.com/>>. Acesso em: 19 mar. 2023.

[24] BATISTA, Jonathan Jalles Silva. **Desenvolvimento de modelos de machine learning para designação de perícias em Juizados Especiais Federais**. 2020. 65 f. Monografia (Pós-Graduação em Tecnologia da Informação) - Instituto Metrópole Digital/Universidade Federal do Rio Grande do Norte, Natal/RN.

[25] MELO, Carlos. Como lidar com dados desbalanceados? Disponível em: <<https://sigmoidal.ai/como-lidar-com-dados-desbalanceados/>>. Acesso em: 19 mar. 2023.